UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/633,076	08/01/2003	Jennifer Melin	200308666-1	4237

22879 7590 05/05/2008

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER	
DICKERSON, CHAD S	

ART UNIT	PAPER NUMBER
2625	

NOTIFICATION DATE	DELIVERY MODE
05/05/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

JERRY.SHORMA@HP.COM
mkraft@hp.com
ipa.mail@hp.com

Office Action Summary	Application No.	Applicant(s)	
	10/633,076	MELIN ET AL.	
	Examiner	Art Unit	
	Chad Dickerson	2625	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *2/25/2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *01 August 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 2/25/2008 has been entered.

Response to Arguments

2. Applicant's arguments with respect to claims 1-18 have been considered but are moot in view of the new ground(s) of rejection. The Amendment to the claims has necessitated the new ground(s) of rejection. However, the reference of Yeung is still applied in the below rejection. The Examiner would like to address an argument the applicant asserted during the remarks regarding the feature of "*transmitting the markup language code that is associated with the configuration attributes supported by the printing device, from the printing device to the requesting device*". The Applicant asserted that this feature is not taught or disclosed by the Yeung reference. The Examiner respectfully disagrees with this assertion.

The Examiner would like to point the Applicant's attention to his or her own remarks on page 8, first paragraph. Here, the applicant states "*Only when a printer-specific data structure (not the UPDF or UPDSD) is not already stored to a fixed disk on*

the computing equipment is a copy accessed from a printer or internet connection (Yeung col. 10, lines 65-67; col. 11, lines 1-2). No communication between the computing device and the printer occurs prior to printing unless a file is missing." Here, the Examiner observed how the applicant states that the printer communicates with the computing equipment and the computing equipment obtains printer-specific data structure information from the printer. Also, the printer-specific data structure is retained in a universal printer description file which is then disposed within a memory area for access and processing by a printer driver. The same universal printer description file requires strict compliance with the syntax of the XML and with the predetermined hierarchy of data elements and corresponding attributes (of the printer) (see col. 2, line 33 – col. 3, line 41 and col. 10, lines 27 – col. 11, lines 24) . Lastly, with the above mentioned statements, the printer-specific data structure is transmitted to the printer driver when a file is missing and the printer-specific data structure is XML that corresponds to the attributes of the printer. Therefore, based on the statement of the applicant and the cited parts of the Yeung reference, the rejection of the above claim feature in question is maintained.

The Examiner would like to also address another claim limitation asserted as not disclosed. The applicant asserts that the feature of "*identifying markup language code embedded in the printing device associated with the configuration attributes supported by the printing device*" is not taught. The Examiner respectfully disagrees with this assertion.

The Examiner would like to simply state that the computing equipment accesses the EEPROM (132) of the printer to identify the universal printer description file (140) for configuration of the printer driver (14). The universal printer description file is considered as markup language code that is associated with the configuration attributes of the printer. Since this file is stored in the printer, this can also be considered as having the markup language that makes up the file to be embedded in the printer (see col. 5, line 23 – col. 6, line 17). The claim is broadly interpreted as having an identification of the code that is associated with the printing capabilities of the printer and this is performed by the printer driver in the system of Yeung when the universal printer description file is referred to by the driver to create a file that enables the driver to use the local printer. Therefore, the rejection regarding this claim feature is maintained.

The Examiner would like to address the argument that the references do not disclose "*the step of excluding markup language code that is associated with configuration attributes not supported by the printing device*". The Examiner respectfully disagrees with this assertion.

The Examiner would like to pose a question to the applicant regarding this claim feature. In the system, the printer driver obtains the universal printer description file (UPDF) and the UPDSD from the printer's EEPROM. The printer-specific data structure is created from the UPDSD related to the specific functions of a printer. The Examiner would like to know how an element that is not a capability of the printer is not excluded from the UPDF, which contains the printer-specific data file (see col. 10, lines 27-48 and

col. 2, line 33 – col. 3, line 49)? The Examiner clearly believes that if the printer cannot perform such a feature or capability, this feature is clearly excluded from the UPDF, which is used to create a printer-specific data structure for the printer driver to fully utilize the functionality of the printer. Explained in column 10 is an example where the UPDF, which stores the printer-specific data structure, or file, in the EEPROM of the printer. The printer driver of the computer can access this file when initializing the printer driver to utilize the printer. Within this UPDF, the Examiner clearly believes that this file stores attributes or capabilities that is fully functional on the printer and excludes other capabilities that are not available on the printer. It is not the purpose of the invention to include elements that are not unique to the printing device, but things that are unique to the specific printer in the printer-specific data structure (see col. 2, lines 57-67).

In regard to the feature above, the applicant is arguing that the feature is not being met because an active user interface is not changed because of the excluded markup language. The Examiner would like to bring to the Applicant's attention that the limitation of an active user interface being changed based on code exclusion appears nowhere in the claims. With the above arguments and the reasoning above as to why the markup language has to be excluded, the rejection of the claim feature is maintained.

In regards to the feature in claim 9, the applicant believes that the reference of Yeung does not disclose the feature stating "*the step of generating a device configuration interface to display the printing device's configuration attributes by*

including markup language code that is associated with the configuration attributes supported b' the printing device". The Examiner respectfully disagrees with this assertion.

Explained in column 5, lines 1-59 is the above feature. The data of the UPDF is exchanged between the EEPROM of the printer to the memory of the computing equipment in order to initialize the printer driver on the computer (see col. 10, lines 27-48). The UPDF file is utilized to enable the printer driver to provide an interface to the printer according to the capabilities, characteristics and features of the printer. In the cited portion listed in the action (col. 11, lines 18-24), the feature of displaying the device's configuration attributes and having this display be an interface to the printer is performed by the reference of Yeung. Since the reference uses the UPDF to enable the printer driver to be an interface to the printer, the above claim feature is performed.

Some of the other deficiencies in the Yeung reference created by the Amendment to the claims are cured using the new references disclosed below.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-4, 6, 9-11, 13-15, 16 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yeung '798 (US Pat No 6426798) in view of Brossman '921 (US Pub No 2005/0179921) and Tanimoto '219 (US Pub No 2003/0126219).

Re claim 1: Yeung '798 discloses a data structure for printer description file, comprising the steps of:

receiving a request for the printing device's configuration attributes at the printing device and the request is received from a requesting device (i.e. the request or query for the printing device's attributes occurs when a printer driver on the computer (40) accesses a printer-specific data structure on an external printer and compares this data structure to the universal printer data structure definition, which is stored on the requesting or querying computer. The printer-specific data structure or universal data structure, illustrated in figure 3, is a plurality of predetermined data elements used for storing various capabilities supported by one of a plurality of printers; see figs. 1-4 and 6; col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24);

identifying markup language code embedded in the printing device associated with the configuration attributes supported by the printing device (i.e. in the system, the printing attributes are automatically mapped to an XML structure that arranges the printing attributes in a hierarchal order. When using the example of figure 6 to determine if a printer-specific data structure is valid, the attributes are compared to the attributes in the universal printer data structure definition. The comparison involves the

attributes and the markup language associated with the attributes. The computing equipment accesses the EEPROM (132) of the printer to identify the universal printer description file (140) for configuration of the printer driver (14). The universal printer description file is considered as markup language code that is associated with the configuration attributes of the printer. Since this file is stored in the printer, this can also be considered as having the markup language that makes up the file to be embedded in the printer; see figs. 3-6; col. 5, line 23 – col. 6, line 17, col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24); and

transmitting the markup language code that is associated with the configuration attributes supported by the printing device, from the printing device to the requesting device (i.e. when the computer (40) used in the system accesses the printer-specific data structure through a communication line (106) to the printer (50), after it is discovered that the printer-specific data structure is valid, the data structure is sent or transmitted to the computer (40) for the printer driver to correctly communicate with the printer (50) using the printer-specific data structure. Also, during the initialization of the printer driver, the computer may access the memory of the printer to obtain the UPDF and the UPDF is transmitted to the computer; see figs. 1-3 and 6; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

However, Yeung '798 fails to specifically teach making a run-time determination in the printing device of the configuration attributes supported by the printing device.

However, this is well known in the art as evidenced by Brossman '921. Brossman '921 discloses making a run-time determination in the printing device of the

configuration attributes supported by the printing device (i.e. in the use of the information handling system that can be incorporated in the printer, the capabilities of the printer are compared to the selected option by the user. The information handling system makes the determination of the attributes supported by the printer and performs the function of notifying a user the incapability of performing the function or takes the print job and performs the requested function. The function of the printer can happen at run-time since this function occurs when the printer is not in a time of designing the markup language associated with the printer capabilities. Also, the printer capabilities are represented by the XML format in an XML file; see figs. 1, 2 and 4; paragraphs [0015]-[0020], [0023] and [0027]-[0037]).

Therefore, in view of Brossman '921, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of making a run-time determination in the printing device of the configuration attributes supported by the printing device in order to see if the printer capabilities are available (as stated in Brossman '921 paragraph [0036]).

However, the combination of Yeung '798 and Brossman '921 fails to teach the feature of wherein the markup language code can enable an active user interface.

However, this is well known in the art as evidenced by Tanimoto '219. Tanimoto '219 discloses the feature of wherein the markup language code can enable an active user interface (i.e. in the system, the printer is used to send device-setting form data in the HTML or XML form to the requesting device (2), or the client terminal. The client terminal uses the device-setting form data to display a window shown in figure 5 to the

user. This is an example of the device-setting form data in HTML or XML being compiled and used to create an active user interface for the user to interact with; see figs. 1-5; paragraphs [0032]-[0036]).

Therefore, in view of Tanimoto '219, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of wherein the markup language code can enable an active user interface incorporated in the device of Yeung '798, in combination with the features of Brossman '921, in order to have the client terminal decode the device-setting form data and show a general browser display to the user (as stated in Tanimoto '219 paragraph [0034]).

Re claim 2: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, wherein the step of identifying markup language code further comprises the step of excluding markup language code that is associated with configuration attributes not supported by the printing device (i.e. the system recognizes the user interface constraints. This defines the maximum allowance of a certain feature. It is believed that no markup language is in relation to the attribute past the threshold of certain interface constraints. In other words, attributes or capabilities that are beyond the printer's functions are not included in the printer-specific data structure used by the printer driver in the system and therefore, the XML schema or codes related to the attributes are also excluded; see fig. 4; col. 2, line 33 – col. 3, line 49 and col. 8, lines 52-62).

Re claim 3: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, wherein the step of identifying markup language code further comprises the step of identifying markup language code associated with groups of configuration attributes supported by the printing device (i.e. in the system, for every function or attribute that is performed by the printer, a markup language code is associated with the function or attribute. This is illustrated in figures 3 and 4. The printer driver identifies these functions in the system when the printer driver is trying to obtain the correct printer capabilities to communicate correctly to the printer with the printer-specific data structure. The data structure is comprised of XML, which is a markup language; see figs. 3 and 4; col. 5, lines 60-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

Re claim 4: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, wherein the step of identifying markup language code further comprises the step of identifying groups of configurations attributes, wherein each group of configurations is associated with a markup language document (i.e. the universal printer data structure definition (150) is defined in XML and is retained in a file referred to as a Document Type Description (DTD). The DTD is considered as a markup language document; see fig. 2 and 3; col. 11, lines 1-24 and col. 12, lines 1-24).

Re claim 6: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, wherein the step of identifying markup language code further comprises the step of identifying markup language code associated with an individual configuration attribute supported by the printing device (i.e. in the system, for every function or attribute that is performed by the printer, a markup language code is associated with the function or attribute. This is illustrated in figures 3 and 4. The printer driver identifies these functions in the system when the printer driver is trying to obtain the correct printer capabilities to communicate correctly to the printer with the printer-specific data structure. The data structure is comprised of XML, which is a markup language; see figs. 3 and 4; col. 5, lines 60-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

Re claim 9: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, further comprising the step of generating a device configuration interface to display the printing device's configuration attributes by including markup language code that is associated with the configuration attributes supported by the printing device (i.e. the printing device's attributes are displayed on the user interface for the user to choose what desired settings the user would like to take place on a document. These settings are accompanied by the markup language that

are transmitted to the printer driver, so that the printer driver can ensure correct communication with the printer using the same printer-specific data structure described in XML, but display in a format for the user to read and understand. The data of the UPDF is exchanged between the EEPROM of the printer to the memory of the computing equipment in order to initialize the printer driver on the computer (see col. 10, lines 27-48). The UPDF file is utilized to enable the printer driver to provide an interface to the printer according to the capabilities, characteristics and features of the printer.; see col. 5, lines 2-67, col. 10, lines 27-67, col. 11, lines 1-24 and col. 12, lines 1-24).

Re claim 10: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, wherein the step of receiving a request for the printing device's configuration attributes further comprises the step of receiving a request for configuration attributes from a device driver for a printing device (i.e. the request or query for the printing device's attributes occurs when a printer driver on the computer (40) accesses a printer-specific data structure on an external printer and compares this data structure to the universal printer data structure definition, which is stored on the requesting or querying computer. The printer-specific data structure or universal data structure, illustrated in figure 3, is a plurality of predetermined data elements used for storing various capabilities supported by one of a plurality of printers; see figs. 1-4 and 6; col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

Re claim 11: Yeung '798 discloses a data structure for printer description file, comprising:

markup language code stored on the printing device (i.e. the markup language code is stored in the ROM (122) or EEPROM (132), in regards to the data elements that represent the capabilities of the printer. The markup language is structured so that the attributes in the system are associated with certain features; see col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67), the markup language code being configured to describe and update the printing device's configuration attributes (i.e. the markup language is used to describe the different functions and attributes of the printer. The XML used is structured in an arrangement that correlates certain features of the printer with XML code. When the determination is made whether the printer-specific data structure matches the universal printer data structure definition, the system checks to see if there are any additional features not accounted for by the universal printer data structure definition (UPDSD), so that these elements may be added to the UPDSD. This is considered as updating the data structure in order to create a better printer-specific data structure in the future; see fig. 2 and 6; col. 3, lines 9-41; col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24); and

a communication module associated with the printing device (i.e. the communication line (106) is considered as the communication module; see figs. 1 and 2; col. 10, lines 27-67), and the communication module is configured to receive requests for configuration attributes and transmit configuration attributes of the printing device

(i.e. when the computer (40) tries to access the printer (40) by the communication line (106), it queries the printer's EEPROM or ROM in order to request from or query the printer's memory to compare the printer-specific data structure to the universal printer data structure definition. This example is analogous to the computer asking to see the universal printer data structure definition to compare it to the printer-specific data structure to see if it is valid. Also, during the initialization of the printer driver, the computer may access the memory of the printer to obtain the UPDF and the UPDF is transmitted to the computer; see fig. 6; col. 3, lines 9-41; col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

However, Yeung '798 fails to teach the feature of an embedded application in communication with the printing device and integrated into the printing device, wherein the embedded application is configured to make a run-time determination of which markup language code corresponds to supported configuration attributes of the printing device.

However, this is well known in the art as evidenced by Brossman '921. Brossman '921 discloses the feature of an embedded application in communication with the printing device and integrated into the printing device (i.e. the information handling system can be considered as the embedded application in communication with the printing device since it checks information received from the outside with the capability information on the inside of the printer. The system can also be incorporated within the printer; see paragraph [0036]),

wherein the embedded application is configured to make a run-time determination of which markup language code corresponds to supported configuration attributes of the printing device (i.e. in the use of the information handling system that can be incorporated in the printer, the capabilities of the printer are compared to the selected option by the user. The information handling system makes the determination of the attributes supported by the printer and performs the function of notifying a user the incapability of performing the function or takes the print job and performs the requested function. The function of the printer can happen at run-time since this function occurs when the printer is not in a time of designing the markup language associated with the printer capabilities. Also, the printer capabilities are represented by the XML format in an XML file; see figs. 1, 2 and 4; paragraphs [0015]-[0020], [0023] and [0027]-[0037]).

Therefore, in view of Brossman '921, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of an embedded application in communication with the printing device and integrated into the printing device, wherein the embedded application is configured to make a run-time determination of which markup language code corresponds to supported configuration attributes of the printing device in order to see if the printer capabilities are available (as stated in Brossman '921 paragraph [0036]).

However, the combination of Yeung '798 and Brossman '921 fails to teach the feature of wherein the markup language code can enable an active user interface.

However, this is well known in the art as evidenced by Tanimoto '219. Tanimoto '219 discloses the feature of wherein the markup language code can enable an active user interface (i.e. in the system, the printer is used to send device-setting form data in the HTML or XML form to the requesting device (2), or the client terminal. The client terminal uses the device-setting form data to display a window shown in figure 5 to the user. This is an example of the device-setting form data in HTML or XML being compiled and used to create an active user interface for the user to interact with; see figs. 1-5; paragraphs [0032]-[0036]).

Therefore, in view of Tanimoto '219, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of wherein the markup language code can enable an active user interface incorporated in the device of Yeung '798, in combination with the features of Brossman '921, in order to have the client terminal decode the device-setting form data and show a general browser display to the user (as stated in Tanimoto '219 paragraph [0034]).

Re claim 13: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a system, wherein the printing device supports printer control language (PCL) (i.e. the printer can support many languages such as PCL5c or PCL6 which are different variations of PCL; see col. 7, lines 7-21).

Re claim 14: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

However, Yeung '798 fails to teach a system, wherein the markup language code includes HTML code.

However, this is well known in the art as evidenced by Tanimoto '219. Tanimoto '219 discloses a system, wherein the markup language code includes HTML code (i.e. the reference discloses describing device-setting form data in XML or HTML; see paragraph [0007]).

Therefore, in view of Tanimoto '219, it would have been obvious to one of ordinary skill at the time the invention was made to a system, wherein the markup language code includes HTML code in order to have a structured document in the HTML form (as stated in Tanimoto '219 paragraph [0018]).

Re claim 15: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a system, wherein the markup language code includes XML code (i.e. the markup language in this invention is XML; see appendix A on page 26; col. 3, lines 9-41; col. 5, lines 42-67; col. 6, lines 1-17).

Re claim 16: Yeung '798 discloses a data structure for printer description file, comprising:

a printing means for printing (i.e. the printer (40) in the system has a printer engine (131) to cause an output from the printer; see col. 5, lines 35-41);

a markup language code means for describing configuration attributes (i.e. the universal print data structure file (140) is used to describe the configuration attributes or the printer. This is utilized by the printer driver to configure itself to be able to print on the printer using the correct attribute options; see fig. 2-4 and 6; col. 5, lines 42-67; col. 6, lines 1-25; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24), wherein the markup language code means is stored on the printing means (i.e. the ROM (122) or EEPROM (132) stores the universal printer description file (140) and the universal printer data structure definition file (150) on the printer (40); see col. 5, lines 42-67; col. 6, lines 1-25); and

a communication module means in the printing means (i.e. the communication line (106) is considered as the communication module; see figs. 1 and 2; col. 10, lines 27-67), wherein the communication port means is for receiving requests for the configuration attributes and transmits configuration attributes supported by the device (i.e. when the computer (40) tries to access the printer (40) by the communication line (106), it queries the printer's EEPROM or ROM in order to request from or query the printer's memory to compare the printer-specific data structure to the universal printer data structure definition. This example is analogous to the computer asking to see the universal printer data structure definition to compare it to the printer-specific data structure to see if it is valid. Also, during the initialization of the printer driver, the computer may access the memory of the printer to obtain the UPDF and the UPDF is

transmitted to the computer; see fig. 6; col. 3, lines 9-41; col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

However, Yeung '798 fails to teach an embedded application means stored in the printing means, wherein the embedded application means is for making a run-time determination of which markup language code corresponds to the configuration attributes supported by the printing means.

However, this is well known in the art as evidenced by Brossman '921. Brossman '921 discloses the feature of an embedded application means stored in the printing means (i.e. the information handling system can be considered as the embedded application in communication with the printing device since it checks information received from the outside with the capability information on the inside of the printer. The system can also be incorporated within the printer; see paragraph [0036]),

wherein the embedded application means is for making a run-time determination of which markup language code corresponds to the configuration attributes supported by the printing means (i.e. in the use of the information handling system that can be incorporated in the printer, the capabilities of the printer are compared to the selected option by the user. The information handling system makes the determination of the attributes supported by the printer and performs the function of notifying a user the incapability of performing the function or takes the print job and performs the requested function. The function of the printer can happen at run-time since this function occurs when the printer is not in a time of designing the markup language associated with the

printer capabilities. Also, the printer capabilities are represented by the XML format in an XML file; see figs. 1, 2 and 4; paragraphs [0015]-[0020], [0023] and [0027]-[0037]).

Therefore, in view of Brossman '921, it would have been obvious to one of ordinary skill at the time the invention was made to have the features of an embedded application means stored in the printing means, wherein the embedded application means is for making a run-time determination of which markup language code corresponds to the configuration attributes supported by the printing means in order to see if the printer capabilities are available (as stated in Brossman '921 paragraph [0036]).

However, the combination of Yeung '798 and Brossman '921 fails to teach the feature of can enable an active user interface.

However, this is well known in the art as evidenced by Tanimoto '219. Tanimoto '219 discloses the feature of can enable an active user interface (i.e. in the system, the printer is used to send device-setting form data in the HTML or XML form to the requesting device (2), or the client terminal. The client terminal uses the device-setting form data to display a window shown in figure 5 to the user. This is an example of the device-setting form data in HTML or XML being compiled and used to create an active user interface for the user to interact with; see figs. 1-5; paragraphs [0032]-[0036]).

Therefore, in view of Tanimoto '219, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of can enable an active user interface incorporated in the device of Yeung '798, in combination with the features of Brossman '921, in order to have the client terminal decode the device-setting

form data and show a general browser display to the user (as stated in Tanimoto '219 paragraph [0034]).

Re claim 18: Yeung '798 discloses a data structure for printer description file, comprising:

a computer usable medium having computer readable program code embodied therein for dynamically controlling access to configuration attributes for a printing device (i.e. the EEPROM (132) has reprogrammable memory that stores information that my be provided to the computing equipment (40) to inform the computer of the operational parameters of the printer (40); see col. 5, lines 23-59), the computer readable program code means in the article of manufacture comprising:

computer readable program code for receiving a request for the printing device's configuration attributes (i.e. the request or query for the printing device's attributes occurs when a printer driver on the computer (40) accesses a printer-specific data structure on an external printer and compares this data structure to the universal printer data structure definition, which is stored on the requesting or querying computer. The printer-specific data structure or universal data structure, illustrated in figure 3, is a plurality of predetermined data elements used for storing various capabilities supported by one of a plurality of printers; see figs. 1-4 and 6; col. 5, lines 42-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24);

computer readable program code for identifying markup language code associated with the configuration attributes supported by the printing device (i.e. in the

system, the printing attributes are automatically mapped to an XML structure that arranges the printing attributes in a hierarchal order. When using the example of figure 6 to determine if a printer-specific data structure is valid, the attributes are compared to the attributes in the universal printer data structure definition. The comparison involves the attributes and the markup language associated with the attributes; see figs. 3-6; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24); and

computer readable program code for transmitting the markup language code that is associated with the configuration attributes supported by the printing device to the requesting device (i.e. when the computer (40) used in the system accesses the printer-specific data structure through a communication line (106) to the printer (50), after it is discovered that the printer-specific data structure is valid, the data structure is sent or transmitted to the computer (40) for the printer driver to correctly communicate with the printer (50) using the printer-specific data structure. Also, during the initialization of the printer driver, the computer may access the memory of the printer to obtain the UPDF and the UPDF is transmitted to the computer; see figs. 1-3 and 6; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

However, Yeung '798 fails to teach the feature of computer readable program code to operate on the printing device for making a run-time determination of configuration attributes supported by the printing device (i.e. in the use of the information handling system that can be incorporated in the printer, the capabilities of the printer are compared to the selected option by the user. The information handling system makes the determination of the attributes supported by the printer and performs

the function of notifying a user the incapability of performing the function or takes the print job and performs the requested function. The function of the printer can happen at run-time since this function occurs when the printer is not in a time of designing the markup language associated with the printer capabilities. Also, the printer capabilities are represented by the XML format in an XML file. It is understood that the feature of the information handling system is utilized through program code that can be operated on the printer in the system since most software functions in a printer are operated in program code executed by the CPU of the printer device; see figs. 1, 2 and 4; paragraphs [0015]-[0020], [0023] and [0027]-[0037]).

Therefore, in view of Brossman '921, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of computer readable program code to operate on the printing device for making a run-time determination of configuration attributes supported by the printing device in order to see if the printer capabilities are available (as stated in Brossman '921 paragraph [0036]).

However, the combination of Yeung '798 and Brossman '921 fails to teach the feature of wherein the markup language code can enable an active user interface.

However, this is well known in the art as evidenced by Tanimoto '219. Tanimoto '219 discloses the feature of wherein the markup language code can enable an active user interface (i.e. in the system, the printer is used to send device-setting form data in the HTML or XML form to the requesting device (2), or the client terminal. The client terminal uses the device-setting form data to display a window shown in figure 5 to the user. This is an example of the device-setting form data in HTML or XML being

compiled and used to create an active user interface for the user to interact with; see figs. 1-5; paragraphs [0032]-[0036]).

Therefore, in view of Tanimoto '219, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of wherein the markup language code can enable an active user interface incorporated in the device of Yeung '798, in combination with the features of Brossman '921, in order to have the client terminal decode the device-setting form data and show a general browser display to the user (as stated in Tanimoto '219 paragraph [0034]).

5. Claim 5 is rejected under 35 U.S.C. 103(a) as being unpatentable over Yeung '798, as modified by Brossman '921 and Tanimoto '219, as applied to claim 1 above, and further in view of Hammond '067 (US Pat No 6820067) and Garcia '470 (US Pub No 2003/0048470).

Re claim 5: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 teaches a method, further comprising the steps of parsing an XML tree containing the printing device's configuration attributes (i.e. the DTD file created using the universal printer data structure definition forms a tree-like structure illustrated in figures 3 and 4. This structure is analyzed, or parsed, to find corresponding printing attributes for the printer-specific data structure used to configure the printer driver in the computer (40); see figs. 1-4 and 6; col. 5, lines 60-67; col. 6, lines 1-17; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24) and using the XML tree to display the

printing device's configuration attributes (i.e. the printing device's attributes are displayed on the user interface for the user to choose what desired settings the user would like to take place on a document. Since the hierarchal structure of the XML code is used in a UPDF and the UPDF is utilized to initialize the printer driver to create an interface for the printer and the printer's capabilities, the feature of an XML tree used to display the printing device's configuration attributes are performed; see col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

However, Yeung '798 fails to teach using the XML tree to create an HTML page that displays the printing device's configuration attributes.

However, this is well known in the art as evidenced by Hammond '067. Hammond '067 discloses using the XML tree to create an HTML page (i.e. the reference discloses that an XML file generated by a compiler (28) is read and produced into a set of HTML web pages; see col. 4, lines 18-32).

Therefore, in view of Hammond '067, it would have been obvious to one of ordinary skill at the time the invention was made to create an HTML page incorporated in the device of Yeung '798, as combined with the features of Brossman '921 and Tanimoto '219, in order to have HTML web pages produced from XML documents (as stated in Hammond '067 col. 4, lines 18-32).

However, the combination of Yeung '798, Brossman '921, Tanimoto '219 and Hammond '067 fails to teach the feature of an HTML page that displays the printing device's configuration attributes.

However, this is well known in the art as evidenced by Garcia '470. Garcia '470 discloses the feature of an HTML page that displays the printing device's configuration attributes (i.e. the printer web page (202) is page that displays the printer settings (222), toner function (224) and status (230) of the printer. The web page performs the function of displaying the printing device's configuration attributes; see paragraphs [0016], [0021] and [0031]-[0038]).

Therefore, in view of Garcia '470, it would have been obvious to one of ordinary skill at the time the invention was made to have the feature of an HTML page that displays the printing device's configuration attributes incorporated in the device of Yeung '798, as combined with the features of Brossman '921, Tanimoto '219 and Hammond '067, in order to have a web page provide access to features of a printer (as stated in Garcia '470 paragraph [0033]).

6. Claims 7, 8, 12 and 17 rejected under 35 U.S.C. 103(a) as being unpatentable over Yeung '798, as modified by the features of Brossman '921 and Tanimoto '219, as applied to claims 1, 11 and 16 above, and further in view of Garcia '470.

Re claim 7: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 teaches a method, wherein the step of receiving a request for the printing device's configuration attributes further comprises the step of receiving the request for the printing device's configuration attributes from a network browser into a printing

device over a network (i.e. with the universal print data structure definition file or the universal print describing file being accessed over the internet or LAN, while the user may select desired printing options through a display on the computer 40), this all is analogous to receiving a request for the printing device's attributes from a network browser into a printing device over a network; see fig. 6; col. 10, lines 27-67; col. 11, lines 1-24 and col. 12, lines 1-24).

However, Yeung '798 fails to teach receiving requests from a network browser into printing device's embedded web server.

However, this is well known in the art as evidenced by Garcia '470. Garcia '470 discloses receiving requests from a network browser into printing device's embedded web server (i.e. the printer web page is accessible from a computer workstation (106) through a browser over network (108). The browser is connected to the web page of the printing device's embedded web server (120), which produces the web site; see paragraphs [0021]-[0026] and [0030]-[0038]).

Therefore, in view of Garcia '470, it would have been obvious to one of ordinary skill at the time the invention was made have the feature receiving requests from a network browser into printing device's embedded web server in order to provide access to the features of the printer (as stated in Garcia paragraph [0033]).

Re claim 8: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

Yeung '798 discloses a method, further comprising the step of using a local area network or the World Wide Web of the Internet as the network (i.e. accessing the printer (40) can be performed through an internet connection or over a local or wide are network; see col. 11, lines 1 and 2).

Re claim 12: The teachings of Yeung '798 in view of Hansen '014 are disclosed above.

However, Yeung '798 fails to teach a system, wherein the communication module is an embedded web server.

However, this is well known in the art as evidenced by Garcia '470. Garcia '470 discloses the communication module is an embedded web server (i.e. in the system, web server is embedded in the printing device, which communicates with other devices on the network through a hosted web page; see paragraphs [0021]-[0026] and [0030]-[0038]).

Therefore, in view of Garcia '470, it would have been obvious to one of ordinary skill at the time the invention was made to the communication module is an embedded web server in order to provide access to the features of the printer (as stated in Garcia paragraph [0033]).

Re claim 17: The teachings of Yeung '798 in view of Brossman '921 and Tanimoto '219 are disclosed above.

However, Yeung '798 fails to teach a system, wherein the communication module means is an embedded web server.

However, this is well known in the art as evidenced by Garcia '470. Garcia '470 discloses a system, wherein the communication module means is an embedded web server (i.e. in the system, web server is embedded in the printing device, which communicates with other devices on the network through a hosted web page; see paragraphs [0021]-[0026] and [0030]-[0038]).

Therefore, in view of Garcia '470, it would have been obvious to one of ordinary skill at the time the invention was made to have wherein the communication module means is an embedded web server in order to provide access to the features of the printer (as stated in Garcia paragraph [0033]).

Conclusion

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

8. Moore '831 discloses driverless printing that discloses the features of having a computer request the attributes of a printer, the printer performing a determination of the attributes present on the apparatus and sending the attributes over to the computer in order to configure the printer driver on the computer for correctly printing on the printer.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHAD DICKERSON whose telephone number is (571)270-1351. The examiner can normally be reached on Mon. thru Thur. 9:00-6:30 Fri. 9:00-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Twyler Haskins can be reached on (571)-272-7406. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/C. D./
/Chad Dickerson/
Examiner, Art Unit 2625

/Twyler L. Haskins/
Supervisory Patent Examiner, Art Unit 2625